Chapter 16

SFDU Usage

The SFDU Usage Standard defines restrictions on the use of Standard Formatted Data Units (SFDUs) in archive quality data sets. **PDS does not require that data products are packaged as SFDUs. However, if data products are packaged as SFDUs, the following standards are in effect.**

A recommendation for the standardization of the structure and construction rules of SFDUs for the interchange of digital space-related data has been prepared by the Consultative Committee for Space Data Systems (CCSDS). An SFDU is a *type-length-value* object. More simply stated, each SFDU consists of a type identifier which indicates the type of data within the SFDU, a length field which either states the length of the data or indicates how the data are delimited, and a value field which is the data itself. Both the type and the length fields are included in a 20 byte label which will be called an *SFDU label* in this document. The value field immediately follows the 20 byte SFDU Label. For PDS data products, the value field contains the PDS label including one or more data object definitions (such as an image).

There are three versions of SFDUs. In Version 1, the length of an SFDU was represented in binary. In Version 2, the length could also be represented in ASCII. In Version 3, the length can be represented in binary, ASCII, or using one of several delineation techniques. Unless previously negotiated, all PDS data products packaged as SFDUs shall be constructed using Version 3 SFDU Labels.

A Version 3 SFDU label consists of the following parts:

l)	Control Authority ID	4 Bytes
2)	Version ID	1 Byte
3)	Class ID	1 Byte
4)	Delimiter Type	1 Byte
5)	Spare	1 Byte
6)	Description Data Unit ID	4 Bytes
7)	Length	8 Bytes

The Control Authority ID and the Description Data Unit ID together form an identifier called an Authority and Description Identifier which points to a semantic (Planetary Science Data Dictionary) and syntactic (Object Definition Language, 2.0) description of the value field.

Version 3 allows delimitation of SFDUs by end-of-file or by start markers and end markers rather than by explicit byte counts. Further details of the SFDU architecture will not be discussed here. Other sources of information can be found in the *SFDU References* listed in the *Introduction* to this document.

Since archive quality data sets are internally defined, only a limited set of SFDU labels are used to identify the files on a data volume. The full suite of available SFDU classes is not used in the packaging of PDS data products. The PDS has adopted this philosophy in order to simplify not only the archive products themselves, but also the software processing of those products. PDS labels are included in the data products, and the information in these PDS Labels is considered more than adequate for data identification and scientific analysis.

The standard usage of SFDUs by PDS in current missions and data restoration is different than the usage of SFDUs in data products from upcoming missions fully supported by the JPL Advanced Multi-Mission Operations System (AMMOS). The following sections define the standard usage of SFDUs for each source of data.

Two SFDU organizations are allowed in PDS data products. The first organization (the ZI Structure) has been used historically in PDS data products from restoration and past missions. The second organization (the ZKI organization) is required for data products which pass through the JPL Advanced Multi-Mission Operations System (AMMOS) Project Database.

16.1   The ZI SFDU Organization

Any PDS data products that are packaged as SFDUs and are not required to pass through the AMMOS Project Database as part of an active mission may use the following SFDU organization.

Each instance of a data product (file) in a data set shall include two (and only two) SFDU labels. These are a Z Class SFDU label and an I Class SFDU label. The two SFDU labels are concatenated (i.e. Z, then I) and left justified in the first line or record of the PDS label for each data product.   (See Figure 16.1.) In the case of data products with detached PDS labels, the two SFDU labels shall appear in the first record of the PDS label files and no SFDU labels appear in the data object files. (See Figure 16.2.)
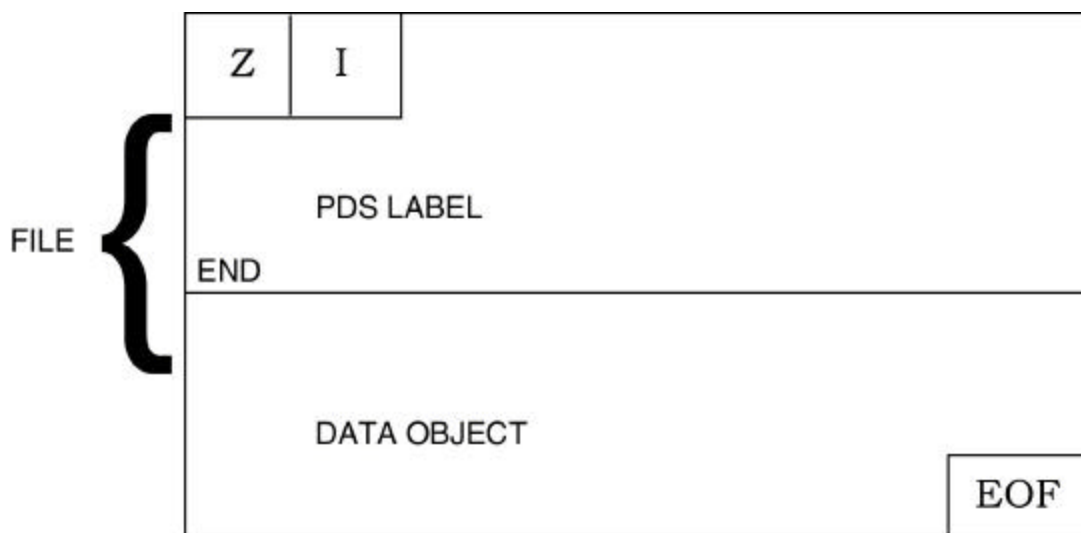

Figure 16.1 Attached PDS Label Example for non-AMMOS compatible products
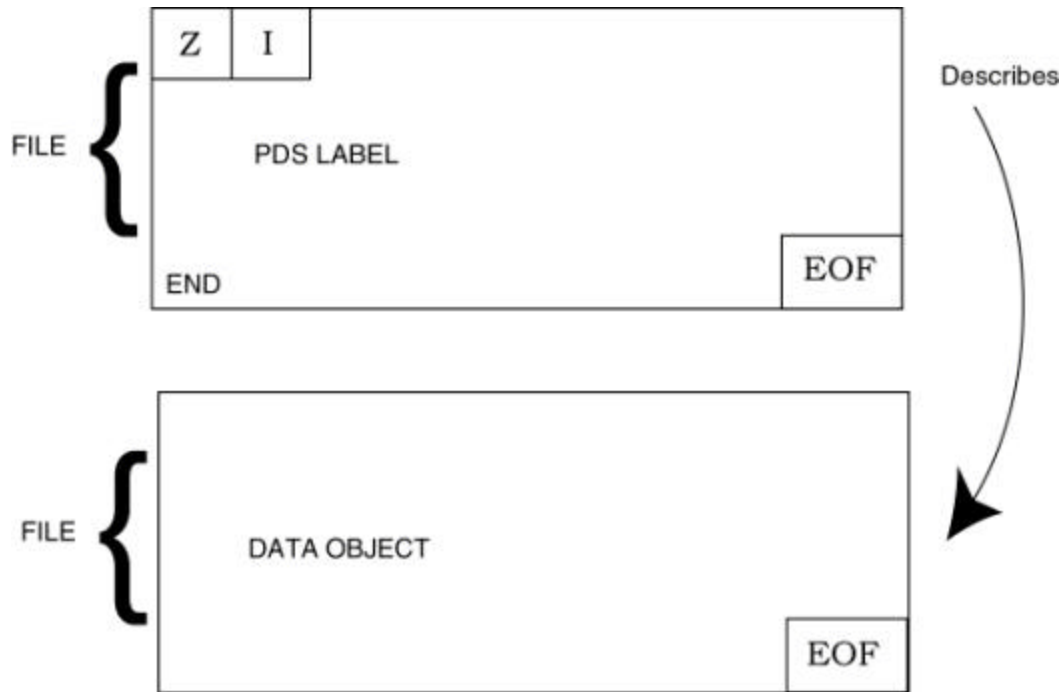
Figure 16.2 Detached PDS Label Example for non-AMMOS compatible products

The first SFDU label shall be a Z Class Version 3 SFDU label. The Z Class indicates that the value field (everything after the first 20 bytes) is an aggregation. In this case, the aggregation consists of only the I Class SFDU. This label also indicates that the delimiter type is End-of-File and that this SFDU (data product) is terminated by a single End-of-File. It shall be formed as follows:

1)	Control Authority ID	CCSD
2)	Version ID	3
3)	Class ID	Z
4)	Delimiter Type	F
5)	Spare	0
6)	Description Data Unit ID	0001
7)	Length Field	00000001

Example: CCSD3ZF0000l00000000l

The second SFDU label shall be an I Class Version 3 SFDU label. Class I indicates that the value field (everything after the second 20 bytes) is application data, the PDS label and the data object(s). The Data Description Unit ID of PDSX indicates that the data product uses the Object Description Language (ODL) syntax and the Planetary Science Data Dictionary semantics to

present data descriptive information. This SFDU label also indicates that the SFDU (data products) will be terminated by a single End-of-File. It shall be formed as follows:

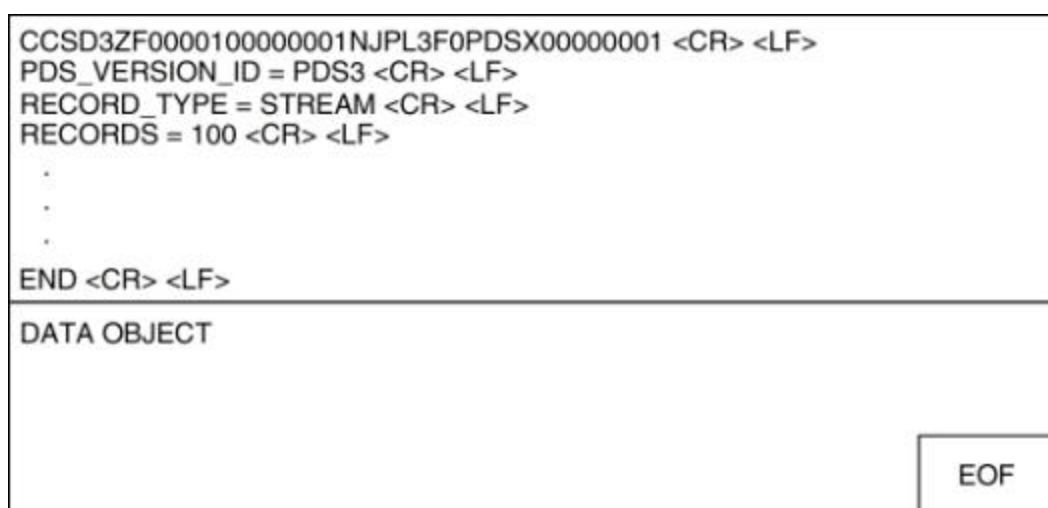|		
1)	Control Authority ID	NJPL
2)	Version ID	3
3)	Class ID	I
4)	Delimiter Type	F
5)	Spare	0
6)	Description Data Unit ID	PDSX
7)	Length Field	00000001

Example: NJPL3IF0PDSX0000000l

```
CCSD3ZF0000100000001NJPL3F0PDSX00000001 <CR> <LF>
PDS_VERSION_ID = PDS3 <CR> <LF>
RECORD_TYPE = STREAM <CR> <LF>
RECORDS = 100 <CR> <LF>

     .

     .

     .

END <CR> <LF>

DATA OBJECT




                                                          EOF
```

Figure 16.3: SFDU Example

The two SFDU labels shall be concatenated, left justified, in the first line or record of the PDS label. Note that there are no characters between the two SFDU labels. See Figure 16.3.

For RECORD_TYPE = STREAM or FIXED_LENGTH or UNDEFINED, the concatenated SFDU labels shall be followed immediately by <CR><LF>. For data products that have RECORD_TYPE =VARIABLE_LENGTH, the two SFDU labels shall not be followed by <CR><LF>.

STREAM example	CCSD3ZF0000l0000000lNJPL3IF0PDSX0000000l <CR><LF>
FIXED_LENGTH Example	CCSD3ZF0000l0000000lNJPL3IF0PDSX0000000l<CR><LF>
VARIABLE_LENGTH Example	CCSD3ZF0000l0000000lNJPL3IF0PDSX0000000l
UNDEFINED Example	CCSD3ZF0000l0000000lNJPL3IF0PDSX0000000l<CR><LF>

The remainder of the PDS label begins on the next line or record. The last line of the PDS label contains the END statement. Then, if the PDS Label is attached, the data object begins on the next record. If the PDS label is detached, the END statement is the last line of the file.

16.2   The ZKI SFDU Organization

Any PDS data products that are packaged as SFDUs and are required to pass through the
AMMOS Project Database as part of an active mission must use the following SFDU
organization. All data products of this type are assumed to have attached PDS labels.

Each instance of a data product (file) in a data set shall include four (and only four) SFDU
labels. These are the Z Class SFDU label, the K Class SFDU label, the End-Marker label for the K
Class SFDU, and the I Class SFDU label. The Z and K Class SFDU labels are concatenated (i.e.
Z, then K) and left justified in the first line or record of the PDS label for each data product. The
End-Marker for the K Class SFDU label and the I Class SFDU label are right justified on the last
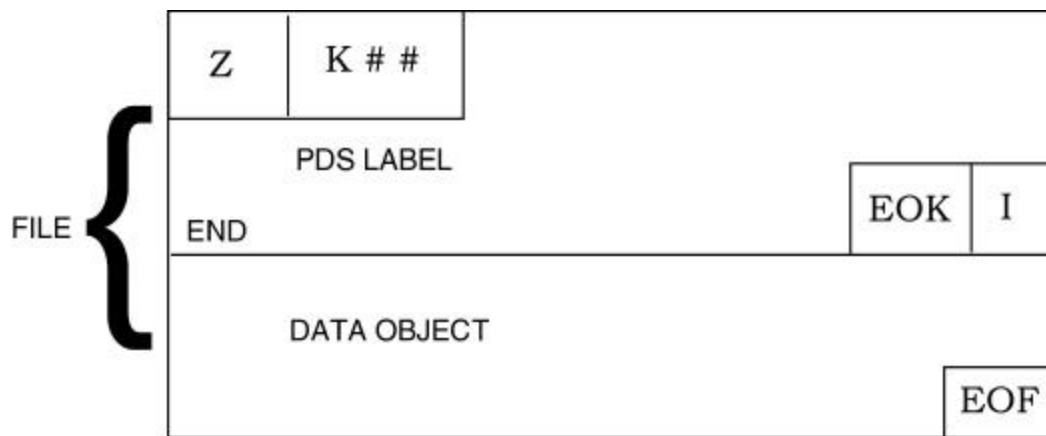record of the PDS label (following the END statement). See Figure 16.4.

Figure 16.4: PDS Label Example for AMMOS compatible products

The first SFDU label shall be a Z Class Version 3 SFDU label. The Z Class indicates that the
value field (everything after the first 20 bytes) is an aggregation. In this case, the aggregation
consists of a K Class (PDS label) and an I Class (data object) SFDU. This label also indicates
that the delimitation type is End-of-File and that this SFDU (data product) is terminated by a
single End-of-File. It shall be formed as follows:

1)	Control Authority	CCSD
2)	Version ID	3
3)	Class ID	Z
4)	Delimiter Type	F
5)	Spare	0
6)	Description Data Unit ID	0001
7)	Length Field	00000001

Example: CCSD3ZF0000l00000000l

The second SFDU label shall be an K Class Version 3 SFDU label. Class K indicates that the
value field (everything after the second 20 bytes) is catalog and directory information, i.e., the
PDS label (sometimes referred to as the K Header). The Data Description Unit ID of PDSX

indicates that the PDS label uses the Object Description Language (ODL) syntax and the Planetary Science Data Dictionary semantics to present data descriptive information. The SFDU label also indicates that the SFDU is delimited by a Start-Marker/End-Marker pair. It shall be formed as follows:

1)	Control Authority ID	NJPL
2)	Version ID	3
3)	Class ID	K
4)	Delimiter Type	S
5)	Spare	0
6)	Description Data Unit ID	PDSX
7)	Length Field	##mark##

The marker pattern (##mark## in the example) can be set to any String which is unlikely to be repeated elsewhere in the data product.

EXAMPLE: NJPL3KS0PDSX##mark##

The two SFDU labels shall be concatenated, left justified, in the first line or record of the PDS label. Note that there are no characters between the two SFDU labels. For data products that have RECORD_TYPE equal to VARIABLE_LENGTH the two concatenated SFDU labels shall not be followed by <CR><LF>.

EXAMPLE: CCSD3ZF0000l0000000lNJPL3KS0PDSX##mark##

The remainder of the PDS label begins on the next line. The last line of the PDS label contains the END statement. Then, in the same line or record, right justified, is the End-Marker for the K Class SFDU and the I Class SFDU label. The End-Marker pattern shall appear as:

EXAMPLE: CCSD$$MARKER##mark##

Note that the start marker and the end marker fields must be identical within the SFDU (in the example, ##mark##). Next shall be an I Class Version 3 SFDU label. Class I indicates that the value field (everything after the SFDU label) is application data, the data object. The Data Description Unit ID varies by data product type, is supplied by the JPL Control Authority, and is usually documented in the science data product Software Interface Specifications (SIS). The SFDU label also indicates that the SFDU will be terminated by a single End-of-File. It shall be formed as follows:

1)	Control Authority ID	NJPL
2)	Version ID	3
3)	Class ID	I
4)	Delimiter Type	F
5)	Spare	0
6)	Description Data Unit ID	XXXX
7)	Length Field	00000001

EXAMPLE: NJPL3IF0010600000000l

where XXXX has been replaced by 0106.

The two SFDU labels shall be concatenated, right justified, and appear in the last line or record of the PDS label following the END statement. (If it happens that there is not 40 bytes left in the last record of the PDS label, add an additional record and right justify the two SFDU labels.) Note that there are no characters between the two SFDU labels, and that the marker pattern and I Class SFDU Labels are transparent to the PDS label processing software (the PDS Toolbox).

Example: END   CCSD$$MARKER##mark##NJPL3IF0010600000000l

The data object begins on the next physical record.

• Example for STREAM record type

End Statement blank(s)   End marker   I Class SFDU   End of record

END   CCSD$$MARKER##mark##NJPL3IF0010600000001<CR><LF>

• Example for FIXED_LENGTH record type:

End Statement Terminator   Record Boundary

END <CR><LF> bbbbb CCSD$$MARKER##mark##NJPL3IF0010600000001

• Example for UNDEFINED record type:

                Statement terminator

End Statement
END<CR><LF> CCSD$$MARKER##mark##NJPL3IF0010600000001

• Example for VARIABLE_LENGTH   RECORD_TYPE:

Record Length   END end of statement

END   CCSD$$MARKER##mark##NJPL3IF0010600000001

16.3   Exceptions to this Standard

Software files and document files should not be packaged as SFDUs. Previous versions of the PDS standards expressed the ZI SFDU labels as an ODL statement. The ZI SFDU labels were followed by "= SFDU_LABEL".

EXAMPLE: CCSD3ZF0000100000001NJPL3IF0PDSX00000001 = SFDU_LABEL